

**Claims:**

1. A memory management method for managing a memory of a testing device for testing a data storage device, said method comprising:

5

reading a data block from said data storage device into a block of memory, said memory block comprising a plurality of smaller memory chunks;

for each said memory chunk, maintaining in real time a record of whether  
10 any data stored in said memory chunk is required to be maintained for use by a test program; and

under a condition of said records indicating that the data in all of the  
memory chunks of said data block need not be maintained for use by said test  
15 program, deleting said data block.

2. The method as claimed in claim 1, wherein maintaining a record comprises:

20 for each said memory chunk, maintaining a flag data indicating a status of data contained within said memory chunk.

3. The method as claimed in claim 1, wherein maintaining a record, comprises:

25

maintaining a count of individual said memory chunks, for which the data stored in said memory chunks is required to be maintained.

4. The method as claimed in claim 1, wherein maintaining a record, comprises:

maintaining a count of individual said memory chunks for which data stored in said memory chunks is required to be maintained; and

5        maintaining for each memory chunk, a pointer to a memory block of which said memory chunk forms a part.

5.        The method as claimed in claim 1, wherein maintaining a record, comprises:

10        maintaining for each memory chunk, a pointer to a memory block of which said memory chunk forms a part; and

              maintaining a pointer to a data chunk stored in said memory chunk.

15        6.        The method as claimed in claim 1, further comprising:

              generating a message to delete said data block in response to a signal from said test program that no further testing of said data block is required.

20        7.        A data management method for managing a plurality of data blocks in a memory device for testing of said data by at least one test program, said method comprising:

25        receiving a plurality of said data blocks in said memory device, each said data block comprising a plurality of data chunks;

              setting a plurality of flags for indicating whether each of said data blocks are to be maintained or not maintained for reading by said at least one test program;

30        maintaining individual said data blocks having a corresponding said flag indicating that said data block is to be maintained; and

deleting said data blocks having flags indicating that said data blocks are not to be maintained.

5

8. The method as described in claim 7, wherein each said data block comprises at least one data chunk, said method comprising:

for each said data chunk, maintaining a pointer to a data block from which  
10 said data chunk originates

9. The method as claimed in claim 7, comprising maintaining a flag indicating that a reader application has finished processing a data block.

15 10. The method as claimed in claim 7, wherein each said data block comprises at least one data chunk, said method comprising maintaining a record of a number of data chunks which are currently in use for a said data block.

11. The method as claimed in claim 7, wherein each said data block  
20 comprises at least one data chunk, said method comprising:

maintaining a counter record indicating a number of data chunks in use for a said data block.

25 12. The method as claimed in claim 7, comprising:

maintaining a flag data indicating whether or not a reader application has finished data processing a data block.

30 13. A method of managing a memory for maintaining a plurality of data blocks in a memory device, such that said data blocks are made available to at

least one reader device which reads data from said data blocks for processing by at least one test component , said method comprising:

reading a said data block into said memory;

5

dividing said data block into a plurality of data chunks;

creating a corresponding respective flag for each said data chunk of said data block;

10

initialising said data flag to an "in use" status;

selecting individual data chunks of said data block for reading by said reader device;

15

reading a block pointer of a selected said data chunk, said block pointer pointing to said data block;

20

processing a said data chunk using said at least one test component; and applying a flag setting to said data block from which said data chunk originates said flag setting indicating that said data chunk has been processed.

14. The method as claimed in claim 13, further comprising:

25

for each of a plurality of data chunks of said data block, maintaining a corresponding respective data flag indicating whether said data chunk is in use or not;

30

in response to all of said data flags of said data block adapting a "not in use" status, deleting said data block from said memory.

15. The method as claimed in claim 13, further comprising:

determining whether any further testing of said data block is required; and

5 If no further testing of said data block is required, deleting said whole data  
block

16. A reader component for reading a plurality of data chunks from a  
memory , said reader component comprising respective sub components for:

10 creating flags for a plurality of data chunks, said flags indicating whether  
each said data chunk is in use or not in use;

15 maintaining a data block flag, said data block flag indicating whether the  
said data block is in use or not;

determining whether said reader device has finished reading from a said  
data block; and

20 generating a signal for deleting a data block which said reader component  
has finished reading from.

17. The reader component as claimed in claim 16, further comprising a  
sub component for generating a message to delete a data block from said  
25 memory.

18. A memory management means for managing a plurality of data  
blocks in a memory , said memory management means comprising:

30 means for effecting receipt of a plurality of said data blocks in said  
memory , each said data block comprising a plurality of data chunks;

means for setting a plurality of data block flags, indicating whether each of said data blocks are in use or not in use by at least one program; and

means for determining whether said data blocks are to be maintained in  
5 said memory or not, depending on a status of a corresponding said flag indicating that said data block is in use, or is not in use by said at least one program.

19. The memory management means as claimed in claim 18,  
comprising:

10

means for maintaining a plurality of data pointers indicating for each of a plurality of data chunks comprising a plurality of data blocks, a data block from which each said data chunk originates

15

20. The memory management means as claimed in claim 18,  
comprising:

means for maintaining a set of flags indicating that a reader application has finished processing a data block.

20

21. The memory management means as claimed in claim 18,  
comprising:

25

means for maintaining a record of a number of data chunks which are currently in use for a said data block.

22. The memory management means as claimed in claim 18,  
comprising:

30

means for maintaining a flag data indicating whether or not a reader application has finished data processing of a data block.

23. The memory management means as claimed in claim 18, further comprising:

means for generating a message to delete a whole said data block from  
5 said memory, if no further testing of said data block is required.

24. A method of memory management for managing a plurality of memory blocks in a memory device for testing data in said memory blocks, said method comprising:

10 partitioning a plurality of said memory blocks in said memory device, each said memory block comprising a plurality of memory chunks each adapted for storing a corresponding respective data chunk;

15 setting a plurality of flags for indicating whether data stored in each of said memory blocks are to be maintained or not maintained; and

20 maintaining data stored in individual said memory blocks having a corresponding said flag indicating that said data of said memory block is to be maintained.

25. A data management method for managing a plurality of data blocks in a memory device, said method comprising:

25 creating a memory block comprising a plurality of memory chunks;

storing a block of data in said memory block, such that individual data chunks comprising said data block are stored in said plurality of memory chunks;

30 maintaining a record of a number of active data chunks in said memory block; and

under conditions where a said active number of data chunks in a memory block is zero, deleting all of said data blocks from said memory block.

5        26.     A memory management method comprising:

reading a data block into a block of memory, said memory block comprising a plurality of smaller memory chunks;

10        for each said memory chunk, maintaining a record of whether data stored in said memory chunk is required to be maintained;

under a condition of the record indicating that data in all of the memory chunks need not be maintained, deleting said data block.

15